MASTER COPY KEEP THIS COPY FOR REPRODUCTION PURPOSES

AD-A251 572

UMENTATION PAGE

Form Approved
OMB No. 0704-0188

2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
April 30, 1992	Interim Tech. Report, July 1990-Sept.199

4. TITLE AND SUBTITLE

A Spatial Similarity Measure for Image Database Applications

5. FUNDING NUMBERS

DAAL03-89-G-0118

6. AUTHOR(S)

V. N. Gudivada, V. V. Raghavan and Dwayne Carr

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Jackson State University, Jackson, MS 39217
University of SW Louisiana, Lafayette, LA 70504

8. PERFORMING ORGANIZATION REPORT NUMBER

DTIC ELECTE JUN 17 1992 S C D

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U. S. Army Research Office
P. O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

ISTRACT *(Maximum 200 words)*

Image Retrieval has been considered as an important task in many application areas such as Geographic Information Systems and Computer-Aided Design. Facilitating retrieval of images based on their similarity to a specified image is a desirous feature of a retrieval scheme for an image database. Providing a suitable means for expressing spatial relationships in a query often improves the ease of specifying it.

In this report, we propose a similarity retrieval algorithm for use in retrieval by spatial similarity. We also describe the generation of a test bed of images and the user interface development. The proposed method has been applied to a test bed of images comprising of floor and furniture layout designs. Each layout design is generated as an image consisting of several objects such as sofa, chair, and table. The dissimilarity between images is based on the notion of distance. The Euclidean distance is computed between the centroids of the matching pairs of constituent objects in both the images. The sum of all such distances plus a suitable penalty for non matching objects is a quantitative measure of spatial similarity. The experimental results obtained using the spatial similarity algorithm quite well agree with our intuitive ranking of the images in the collection.

92-15461

14. SUBJECT TERMS	15. NUMBER OF PAGES
Image Databases, Spatial Databases	13
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

A Spatial Similarity Measure for Image Database Applications*

V. Naidu Gudivada, Vijay V. Raghavan, and Dwayne Carr

Technical Report: 91 - 1

Department of Computer Science
Jackson State University
Jackson, MS 39217

* This research is supported by the Department of Defense under Grant No: DAAL03-89-G-0118.

A Spatial Similarity Measure for Image Database Applications

Abstract

Image Retrieval has been considered as an important task in many application areas such as Geographic Information Systems and Computer-Aided Design. Facilitating retrieval of images based on their similarity to a specified image is a desirous feature of a retrieval scheme for an image database. Various types of retrieval by similarity can be defined based on the domain characteristics and hence they are usually domain dependent. Spatial similarity assesses the degree to which the spatial relationships of objects in an image confirm to those specified by a user query. Providing a suitable means for expressing spatial relationships in a query often improves the ease of specifying it.

In this report, we propose a similarity retrieval algorithm for use in retrieval by spatial similarity. We also describe the generation of a test bed of images and the user interface development. A user specifies a spatial query in its most natural analog form. The proposed method has been applied to a test bed of images comprising of floor and furniture layout designs. Each layout design is generated as an image consisting of several objects such as sofa, chair, and table.

The dissimilarity between images is based on the notion of distance. The Euclidean distance is computed between the centroids of the matching pairs of constituent objects in both the images. The sum of all such distances plus a suitable penalty for non matching objects is a quantitative measure of spatial similarity. The experimental results obtained using the spatial similarity algorithm quite well agree with our intuitive ranking of the images in the collection.

1. Introduction

This report describes the generation of a test bed of images, the user interface development, and the similarity-based retrieval algorithm for use in retrieval by spatial similarity in image database applications [1, 2, 3, 4, 5, 6, 7, 9, 13]. Living room floor and furniture layout design is chosen as the domain of study. We have implemented this project on an IBM Personal Computer (PS/2, Model 70) using Metaware High C compiler [10] and Phar Lap DOS Extender [11].

In many image database environments, retrieval of images based on their similarity to a given image is very desirable. Typically a user specifies such a query by sketching it on the screen using graphical input devices and by assigning attribute properties to the objects contained in the sketched image. This image is referred to as the query image. The query image is input to another module which compares the query image to each image in a database of layout designs for retrieval by spatial similarity. In the domain chosen for the current work, each living room scene is generated as an image consisting of several objects such as sofa, chair, and table. The layouts differ in the number of objects, the types of objects, their geometrical and non-geometrical attributes as well as the spatial orientation of the objects with reference to the other objects in the same layout.

The remainder of the paper is organized as follows. The graphics system called HOOPS which is used to generate the test bed images is briefly discussed in section 2. The user interface for the prototype system developed is described in section 3. The spatial similarity retrieval algorithm is presented in section 4. Finally, section 5 provides experimental results and concludes the report.

2. Image Database of *Furniture Layout Design*

An object-oriented graphics library called HOOPS (Hierarchical Object Oriented Picture System) is used to generate all the images in the collection [8]. The library is designed to be linked with application programs written in high-level programming languages. An image in HOOPS is organized as a hierarchical collection of picture objects called segments. The tree structure allows HOOPS to implement the concept of inheritance. It is a convenient mechanism for lower-level segments to inherit non-geometrical attributes such as color, line style, and visibility from higher-level segments in the hierarchy.

A distinguishing feature of HOOPS is the metafile. Once an image is generated it can be saved in a metafile either for subsequent display without regeneration or to be used as input to other modules in the system. Metafile is an ASCII file and the image is represented in vector format in the metafile. For images which are not overly complex, vector format provides a compact representation over the ubiquitous raster format. A specific set of calls with appropriate geometric descriptions to the HOOPS graphics library generates the desired layout.

A total of twenty-five layouts were generated for testing the spatial similarity retrieval algorithm. Each living room scene is generated as an image consisting of several objects such as sofa, chair, and table. The layouts differ in the number of objects, the

types of objects, their geometrical and non-geometrical attributes as well as the spatial orientation of the objects with reference to the other objects in the same layout. Some sample images of the furniture layout design database are shown in Appendix.

3. User Interface

The user interface (UI) developed for the spatial similarity retrieval provides an intuitive and natural mechanism for users to query the image database. The UI provides interaction primarily through a pointing input device and pull-down menus in a windowing framework. The part of any window which is intended for the system to display results or for the user to sketch a query is referred to as the **sketch area.** When the program is started the **Main Window** appears as shown in Figure 1. This window provides two options: Files and Query. The **Files** option provides browsing facilities and enables interactive sketching of spatial queries and the **Query** option opens a new window called **Similarity Retrieval Window** as shown in Figure 4. Selecting the Files option provides four choices as a sub-menu. The sub-menu items are: List, Close, Create, and Exit (Figure 2).

The **List** option displays symbolic names of all the images in the collection and the user has the option to display a selected image for browsing. The **Close** option clears the sketch area and the **Exit** option takes the control back to the Main Window. The **Create** option opens a new window called **Query Creation Window** as shown in Figure 3. The menu items on the Query Creation window are: Insert, Delete, Position, Rotate, Stretch, and Exit. The **Insert** option displays a list of domain objects for selection by the user and inserts the selected object into the sketch area and the **Delete** option removes a selected object from the sketch area. **Position**, **Rotate**, and **Stretch** options provide means for translation, rotation, and scaling geometrical transformations, respectively. Finally, the **Exit** option takes the user back to the Main Window. In summary, Query Creation Window enables the user to specify a spatial query by selecting any number of domain objects and spatially orienting them to suit his specification needs.

The **Similarity Retrieval Window** is intended to initiate the spatial similarity retrieval calculation for the query specified using the Query Creation Window and to display the results in a browsing mode. The various items on the menu bar are: Start Query, Display Results, Show Query, and Exit (Figure 4). The **Start Query** option simply signals the system to initiate the similarity retrieval calculations. The **Show Query** option displays the query sketched by the user and the **Exit** option takes the control back to the Main Window. The Display Results option opens a new window called **Display Results Window** as shown in Figure 5.

The Display Results Window menu bar provides the following options: Next, Previous, Current, Show Query, and Exit. The image in the database that has the highest similarity to the specified query is shown when this window is initially displayed. The **Next** option displays the image that has the next highest similarity to the specified query and the **Previous** option displays an image that has the immediately next highest similarity relative to the image that is currently being specified. The **Show Query** option displays the specified query for the user to intuitively asses the relevance of a retrieved

query to the specified query. The **Current** option clears the image displayed in response to Show Query option and restores the sketch area. In essence, using Current and Show Query options, the user can switch between the retrieved image and the query image quickly to visually asses their similarity. As usual, the **Exit** option takes the control back to the Main Window.

4. Spatial Similarity Retrieval Algorithm

The similarity retrieval algorithm is based on a notion of distance between two images or between query image and a database image. If the distance between two images is zero, then the images are identical. The greater the distance, greater is the spatial dissimilarity between the images. When an user query is incrementally sketched, the system keeps track of the various objects that have been added to the query and the associated geometric and non-geometric attributes. Specifically, our spatial similarity algorithm uses the following information on each object type: the type of the object, the number of instances of the object, and the coordinates of the centroid of each object instance.

The set of objects contained in a query image is referred to as the **query object set** and the set of objects contained in a database image is referred to as the **candidate object set.** These are actually multisets since each object type can have several instances. The spatial similarity algorithm works as follows. The query image is evaluated for spatial similarity with every image in the database. We now illustrate the process involved in computing the spatial similarity of a query image with an image in the database.

1. For an object type instance in the query object set find a corresponding object type instance in the candidate object set

2. If only a single match is found then associate the matched object instance with the object instance in the query object set. If multiple matches are found then associate one of the matched objects that is spatially closest with the object instance in the query object set. The spatial closeness is based on the Euclidean distance between the corresponding centroids. The smaller this value, the closer are the objects, spatially. In either case mark the matched objects in the candidate object set as not available for further matching.

3. Repeat steps 1 and 2 for each object type instance in the query image

As a result of the above process, for a given pair of query and candidate images, we observe and perform the following:

a. Each instance of each object type in the query object set has either a match or no match.

b. For those instances having a match, compute the Euclidean distance between the corresponding centroids.

c. The sum of all these distances plus suitable penalties for unmatched objects both in the query object set and in the candidate object set gives us a quantitative measure of distance or dissimilarity between images.

d. The penalties to be imposed for unmatched objects is domain dependent. For our implementation, a constant value is added for each unmatched instance.

The algorithm runs in $O(n^2)$ time. The algorithm is sensitive to the order in which the object instances in the query image is matched with the corresponding object instances in the candidate image. The approach to spatial similarity retrieval considered here is more general than that proposed in [12], which assumes that there is only one instance of an image object type in any given image.

Experimental Results

Of the twenty-five layouts generated, we have considered each layout as a query image in turn. For each query image, all the images in the collection are ranked in increasing order of distance measure. The results obtained for each query match our intuitive ranking of the images in the collection. One of our future plans is to make the algorithm insensitive to the order of matching. The algorithm considers only the instances of the same object type during the matching process. However, if this kind of match cannot be found, it is reasonable to match instances of different object types which are semantically related (e.g. two tables having different shape tops). This aspect will also be investigated in the future research.

References

1. Chang, S.K. (1981), Guest Editor, "Special Issue on Pictorial Information Systems," ***IEEE Computer***, No. 11.

2. Chang, S.K., Shi, Q.Y., and Yan, C.W. (1987), "Iconic Indexing by 2D Strings," ***IEEE Transactions on Pattern Analysis and Machine Intelligence***, Vol. 9, No. 3, pp. 413-428.

3. Chang, S.K., Yan, C.W., Dimitrof, D., and Arndt. T. (1988) "An Intelligent Image Database System," ***IEEE Transactions on Software Engineering***, Vol. 14, pp. 681-688.

4. Chang, S.K. (1989), ***Principles of Pictorial Information Systems Design***, Prentice-Hall, Englewood Cliffs, NJ.

5. Grosky, W.I. and Mehrotra, R. (1989a), "Guest Editor's Introduction to Image Database Management," ***IEEE Computer***, Vol. 22, No. 12, pp. 7-8.

6. Guenther, O. and Buchmann, A. (1990), Research Issues in Spatial Database, **SIGMOD RECORD**, Vol. 19, No. 4, pp. 61-68.

7. Iyengar, S.S. and Kashyap, R.L. (1988), Guest Editors, ***IEEE Transactions on Software Engineering***, Special Issue on Image Databases.

8. Ithaca Software, Alameda, CA., HOOPS Graphics System.

9. Lee, S.Y., Shan, M.K. and Yang, W.P. (1989), "Similarity Retrieval of ICONIC Image Database," ***Pattern Recognition***, Vol. 22., No. 6, pp. 675-682.

10. Metaware Inc., Santa Cruz, CA., HIGH C Compiler.

11. Phar Lap Software, Inc. Cambridge, MA., 386/ASM Link Software.

12. Raghavan, V.V and Gudivada, V.N. (1990), "A Domain Independent Similarity Measure for Symbolic Images," ***Indian Computing Congress***, November, pp. 195-203.

13. Tamura, H. and Yokoya, N. (1984), "Image Database Systems: A Survey," ***Pattern Recognition***, Vol. 17, No. 1, pp. 29-43.

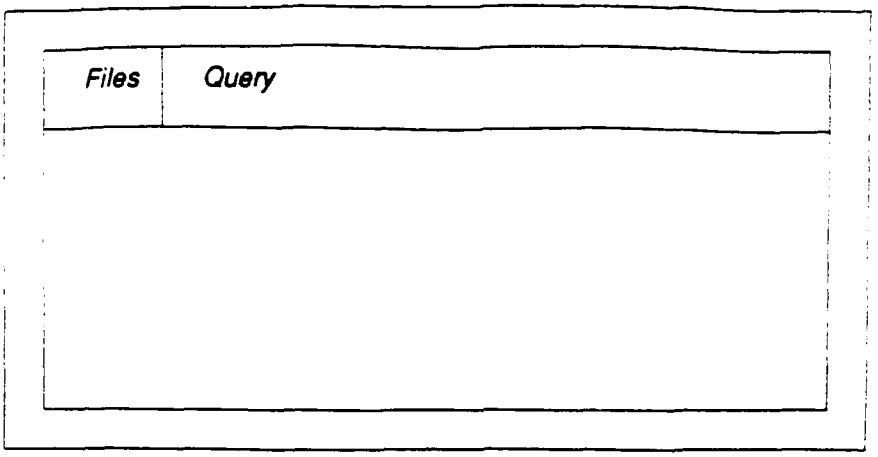

Figure 1: Main Window

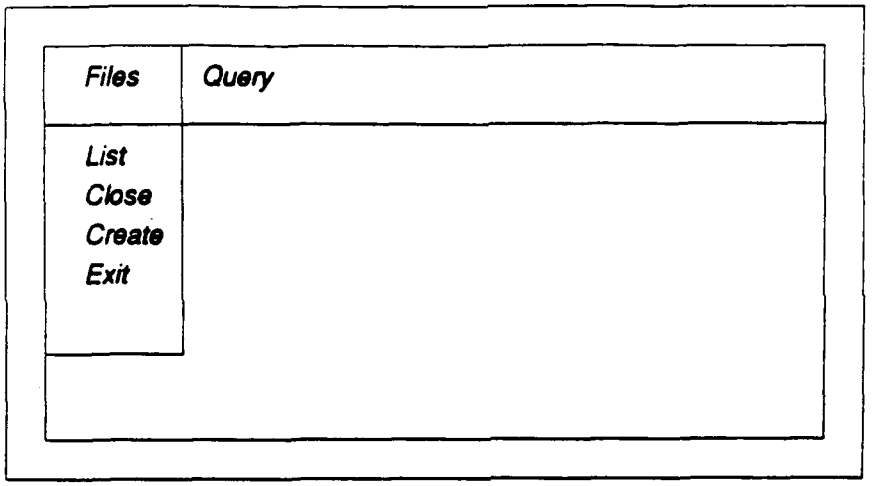

Figure 2: Files option expanded

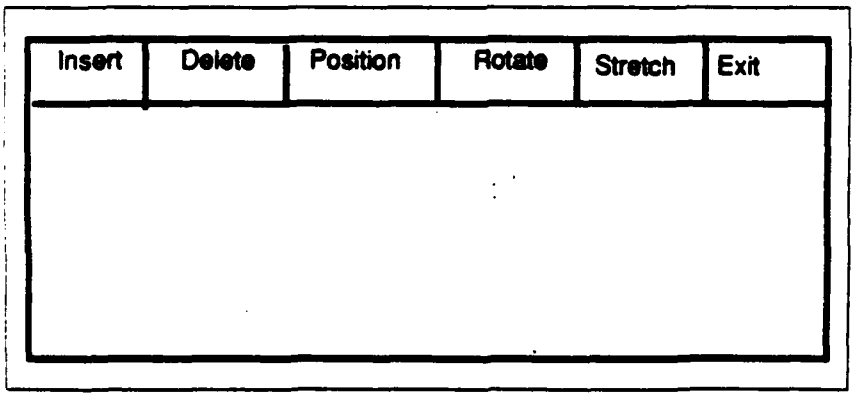

Figure 3: Query Creation Window

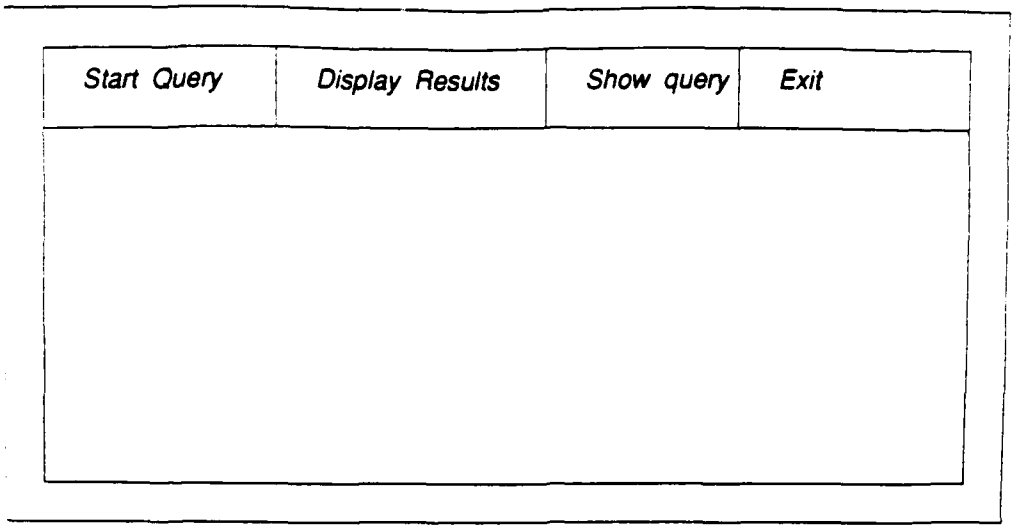

Figure 4: Similarity Retrieval Window

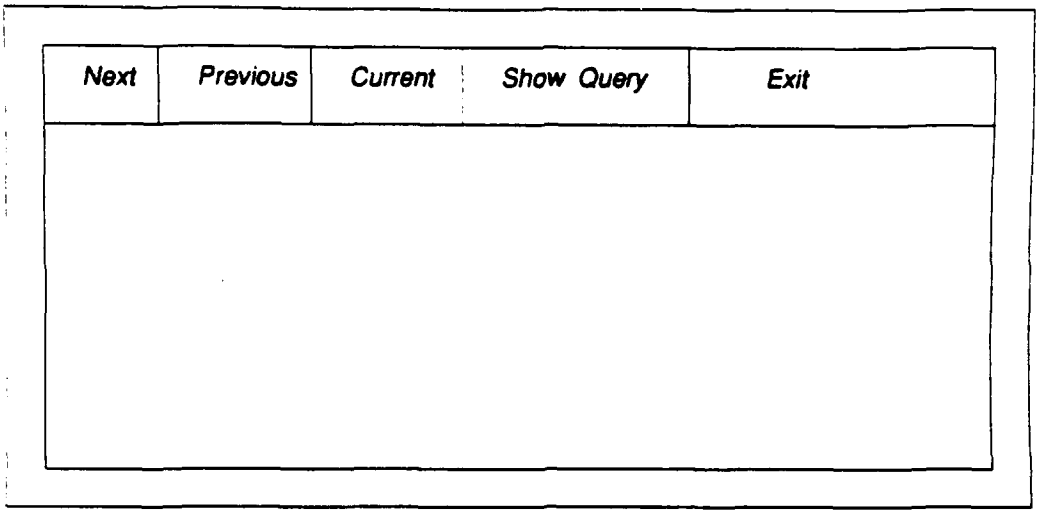

Figure 5: Display Results Window

APPENDIX

Sample Images From *Living Room Layout Design* Database